

(19)



JAPANESE PATENT OFFICE

JPA 8-88775

PATENT ABSTRACTS OF JAPAN

(11) Publication number: **08088775 A**(43) Date of publication of application: **02.04.96**

(51) Int. Cl.

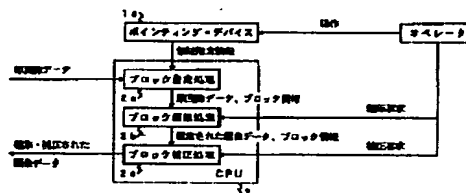
H04N 1/415
G06T 9/00
(21) Application number: **06222303**(71) Applicant: **TOPPAN PRINTING CO LTD**(22) Date of filing: **19.09.94**(72) Inventor: **HONMA TOSHIMITSU**(54) **IMAGE PROCESSING UNIT**

COPYRIGHT: (C)1996,JPO

(57) Abstract:

PURPOSE: To eliminate or reduce distortion caused by image compression processing by applying revise processing to part of a coefficient obtained through discrete cosine transformation in the unit of divided blocks of an image into a desired value.

CONSTITUTION: When a correction request is made as to an area designated by the operator in an image subjected to edit processing and before compression, a CPU 2 uses a block correction processing function 2c to make correction processing. In this correction processing, a picture element subjected to distortion correction is obtained by the processing subtracting a vector (g) obtained by inverse DCT to an original image vector including an allowable error from a vector (y) of image data. The CPU 2 replaces the picture element of an object block with a corrected picture element and applies compression processing for storage or transmission. The distortion between a block area and an adjacent image area is eliminated or suppressed into a permissible error by revising part of coefficients obtained by applying DCT to a block having the distortion.



The Partial English translation of JPA 8-88775

[0007]

A typical example is an image having moderate changes like graduation. Fig. 6 (a) shows an example of changes in tone in an image having moderate changes like graduation. If, for example, an image has been divided into blocks and then compressed by the block, stretching the compressed image in the image decompression process makes the difference in tone of pixel data between adjacent blocks as conspicuous as shown in Fig. 6 (b). And borders between pixels become noticeable as if they are mosaicked. This is the block distortion.

JPA8-88775

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平8-88775

(43) 公開日 平成8年(1996)4月2日

(51) Int. Cl. ⁶

H04N 1/415

G06T 9/00

識別記号

庁内整理番号

F I

技術表示箇所

G06F 15/66

330

H

審査請求 未請求 請求項の数 6 O L (全14頁)

(21) 出願番号 特願平6-222303

(22) 出願日 平成6年(1994)9月19日

(71) 出願人 000003193

凸版印刷株式会社

東京都台東区台東1丁目5番1号

(72) 発明者 本間 俊光

東京都台東区台東一丁目5番1号 凸版印刷株式会社内

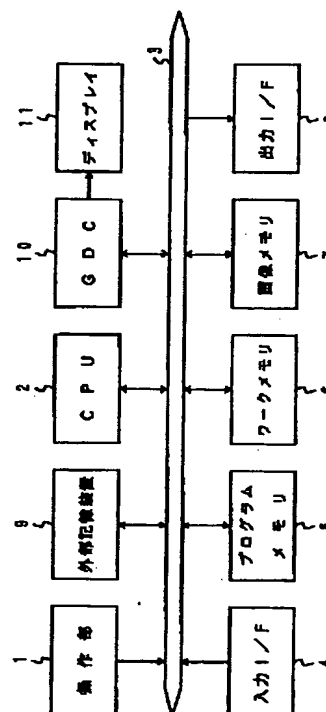
(74) 代理人 弁理士 鈴江 武彦

(54) 【発明の名称】 画像処理装置

(57) 【要約】

【目的】 本発明は、画像編集、もしくは画像圧縮処理を行うことによって生じる歪みを除去もしくは許容範囲内で低減することができるようにする。

【構成】 画像編集、もしくは画像をブロックに分割して各ブロック単位で離散コサイン変換し、画像圧縮処理を行う機能を有する画像処理装置において、画像の分割したブロック単位で離散コサイン変換して得られる係数の所望の一部を所望値に変更処理する処理手段2を設けて構成する。



【特許請求の範囲】

【請求項1】 画像編集、もしくは画像をブロックに分割して各ブロック単位で離散コサイン変換し、画像圧縮処理を行う機能を有する画像処理装置において、画像の分割したブロック単位で離散コサイン変換して得られる係数の所望の一部を所望値に変更処理する処理手段を設けたことを特徴とする画像処理装置。

【請求項2】 処理手段には、係数変更の際に、直流成分に対応する係数及び交流成分のうちの最も低い周波数を持つ基底関数に対応する係数を所望値に変更処理する機能を持たせることを特徴とする請求項1記載の画像処理装置。

【請求項3】 画像編集、もしくは画像を画像ブロックに分割して各ブロック単位で離散コサイン変換し、画像圧縮処理を行う機能を有する画像処理装置において、上記画像ブロック内を所望の小ブロックに分割すべく指定すると共に、上記画像ブロックの境界領域を含む一つ以上の小ブロックを指定するための手段と、上記画像ブロック単位で離散コサイン変換処理すると共に、上記指定された各小ブロックと、該画像ブロックに隣接するブロックに対しては上記離散コサイン変換処理後の得られる係数の所望の一部を所望値に変更処理する処理手段とを設けたことを特徴とする画像処理装置。

【請求項4】 画像編集、もしくは画像を画像ブロックに分割して各ブロック単位で離散コサイン変換し、画像圧縮処理を行う機能を有する画像処理装置において、画像の局所領域とその領域における所望の編集要求を指定する手段と、画像を前記指定された局所領域に従って分割する第1の機能、この第1の機能により分割された領域内の画像について上記画像ブロック単位で離散コサイン変換処理すると共に、上記指定された各小ブロックと、該画像ブロックに隣接するブロックに対しては上記離散コサイン変換処理後の得られる係数の所望の一部を所望値に変更処理する第2の機能とを有する処理手段とを設けたことを特徴とする画像処理装置。

【請求項5】 画像を画像ブロックに分割して各ブロック単位で離散コサイン変換し、画像圧縮処理を行う機能を有する画像処理装置において、画像の領域を指定すると共に、その領域での圧縮処理内容を指定する手段と、画像を画像ブロックに分割して各ブロック単位で離散コサイン変換し、画像圧縮処理すると共に、上記指定された領域では各ブロック単位で離散コサイン変換し、上記指定された圧縮処理内容の圧縮処理を行い、その後、上記指定された領域内の各ブロックと、該ブロックに隣接するブロックに対しては上記離散コサイン変換処理後の得られる係数の所望の一部を所望値に変更処理する処理手段とを設けることを特徴とする画像処理装置。

【請求項6】 処理手段には、画像を前記指定された局

所領域に従って分割する第1の機能、この第1の機能により分割された領域内の画像について上記画像ブロック単位で離散コサイン変換処理すると共に、上記指定された各小ブロックと、該画像ブロックに隣接するブロックに対しては上記離散コサイン変換処理後の得られる係数の所望の一部を所望値に変更処理する第2の機能とを設けることを特徴とする請求項5記載の画像処理装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、オペレータの指定に応じて画像データを編集することにより、得られた画像を効率的に圧縮して蓄積し、伸張する画像処理装置に関する。

【0002】

【従来の技術】 従来より、画像データを目的に合うように編集するために、種々の機能が開発されており、また、編集の目的も様々である。その中で、編集にあたるオペレータの所望する画像のある領域を、局所的に変更して編集効果を上げたいと云う要求があり、そのような要求に対応するための機能を実現するためのコンピュータ・ソフトウェア（アプリケーション・プログラム）も市販されている。ここで編集要求には、例えば、各種フィルタリングによる“ぼかし”や“シャープネス向上”，“明るさ”や“色の調整”等々、様々なものがある。

【0003】 このような編集を行うと云うことは、元の画像を局所的に変更すると云うことであるから、編集処理の結果、当然のことながら領域の境界に歪みが生じる。しかし、編集後に領域の境界に生じるこのような歪みについての対応は十分になされていないので、不自然な画像になることが避けられない。

【0004】 また一方、画像データ量が膨大であることから、近年においては画像データを扱うにあたり、JPEG（静止画像圧縮の国際標準方式）やMPEG（動画像圧縮の国際標準方式）に見られるような、画像データを圧縮して記憶する方法が発展し、編集した結果を圧縮して記憶するケースが多くなってきている。これらの方式では圧縮率を上げるために、やむをえず画像の劣化を許している。画像圧縮による画像劣化の種類としては、ブロック間の歪みと、モスキート・ノイズがよく知られている。そして、それらの劣化への対応策も幾つか提案されている。

【0005】 ところで、上述のような国際標準方式に用いられる画像圧縮においては、圧縮率が同程度であっても、画像の内容により、画質劣化の程度が異なってくる。そのため、画像を編集した後にある圧縮率を目指して圧縮操作を施す場合の画質劣化は、編集後の画像データに依存し、従って、編集方法にも依存することになる。

【0006】 このように画像データに対する編集操作

は、データ圧縮時の画質に影響するが、従来においては、編集効果と圧縮による画質の両方を向上させることが十分に考慮されていない。

【0007】典型的な例は、グラデーションのようなゆるやかな変化のある画像である。グラデーションのようなゆるやかな変化のある画像部分の階調変化を示すと例えば、図6(a)の如きである。これを例えば、画像のブロック分割をして各ブロック単位で画像圧縮を行うと、画像の復元のために、圧縮伸長した際に隣接するブロック間では画素データの階調差が図6(b)のように顕著なものとなり、あたかもモザイクをかけたように、画素間での境界が顕著なものになってしまう。これがブロック歪みである。

【0008】JPEGやMPEGでは、画像圧縮にあたって、画像をブロックに分割し、各ブロックに直交変換としてDCT（離散コサイン変換）を施し、それによって得られる係数の量子化を圧縮の手法に利用している。そのため、これらJPEGやMPEGでは、圧縮率が高くなるとグラデーションのようなゆるやかな変化のある画像領域では、ブロック歪みが顕著に現われることが避けられない。このような圧縮による画質劣化を低減する手段が編集時に十分に考慮されていない。

【0009】

【発明が解決しようとする課題】画像のある領域を、目的に合うように局所的に変更する画像編集が実施されているが、このような編集を施す画像編集装置は、編集した画像を画像圧縮して保存する。

【0010】一方、画像データは膨大な容量を持つことから、画像データの保存や、伝送などには画像圧縮を施すのが普通であり、画像圧縮のための国際標準方式としてJPEGやMPEGがあることから、データの標準化を図るために画像圧縮にはこれらの国際標準方式を採用する必要がある。

【0011】従って、編集処理を施した画像についても保存や伝送をするにあたっては当然、この国際標準方式を適用して画像圧縮を施す。しかも、データ量をできるだけ少なくするためには、高圧縮率の圧縮処理が要求される。

【0012】しかし、国際標準方式であるJPEGやMPEGでは、画像をブロック分割して直交変換処理し、ブロック内の成分を周波数成分別の係数に分けてから、係数を量子化し、これを圧縮符号化して画像圧縮する。圧縮画像の伸長はこれと逆の手順を踏む。

【0013】ここで圧縮率を高くするとグラデーションのようなゆるやかな変化のある画像領域では、ブロック歪みが顕著に現われることが避けられない。このような圧縮による画質劣化を低減することが、上述の画像編集時に十分に考慮されていない。

【0014】従って、従来の技術では、画像データの編集領域の指定方法、編集による画像データの局所変更後

の境界領域の歪みの低減、及び編集結果の圧縮・伸張におけるブロック歪みの低減と圧縮率の向上に関して、総合的な視点からの解決を与えるものが提供されていないことから、ブロック歪みによる画像の劣化が問題となる。そのため、あまり高い圧縮率を用いることができないと云う問題がある。

【0015】そこで、本発明の目的とするところは、画像データの編集領域の指定方法、編集による画像データの局所の変更後の境界領域の歪みの低減、及び編集結果の低歪みで効率的な圧縮・伸張の各々の課題及びそれらの課題の関連をも考慮した解決手段を持つ画像編集記憶装置を提供することであり、これによって、上述の課題のより良い解決を図るものである。

【0016】

【課題を解決するための手段】上記目的を達成するため、本発明はつぎのように構成する。すなわち、画像編集、もしくは画像をブロックに分割して各ブロック単位で離散コサイン変換し、画像圧縮処理を行う機能を有する画像処理装置において、第1には、画像の分割したブロック単位で離散コサイン変換して得られる係数の所望の一部を所望値に変更処理する処理手段を設ける。また、処理手段には、係数変更の際に、直流成分に対応する係数及び交流成分のうちの最も低い周波数を持つ基底関係に対応する係数を所望値に変更処理する機能を持たせる。

【0017】また、第2には、画像編集、もしくは画像を画像ブロックに分割して各ブロック単位で離散コサイン変換し、画像圧縮処理を行う機能を有する画像処理装置において、上記画像ブロック内を所望の小ブロックに分割すべく指定すると共に、上記画像ブロックの境界領域を含む一つ以上の小ブロックを指定するための手段と、上記画像ブロック単位で離散コサイン変換処理すると共に、上記指定された各小ブロックと、該画像ブロックに隣接するブロックに対しては上記離散コサイン変換処理後の得られる係数の所望の一部を所望値に変更処理する処理手段とを設ける。

【0018】また、第3には、画像編集、もしくは画像を画像ブロックに分割して各ブロック単位で離散コサイン変換し、画像圧縮処理を行う機能を有する画像処理装置において、画像の局所領域とその領域における所望の編集要求を指定する手段と、画像を前記指定された局所領域に従って分割する第1の機能、この第1の機能により分割された領域内の画像について上記画像ブロック単位で離散コサイン変換処理すると共に、上記指定された各小ブロックと、該画像ブロックに隣接するブロックに対しては上記離散コサイン変換処理後の得られる係数の所望の一部を所望値に変更処理する第2の機能とを有する処理手段とを設ける。

【0019】また第4には、画像を画像ブロックに分割して各ブロック単位で離散コサイン変換し、画像圧縮処

理を行う機能を有する画像処理装置において、画像の領域を指定すると共に、その領域での圧縮処理内容を指定する手段と、画像を画像ブロックに分割して各ブロック単位で離散コサイン変換し、画像圧縮処理すると共に、上記指定された領域では各ブロック単位で離散コサイン変換し、上記指定された圧縮処理内容の圧縮処理を行い、その後、上記指定された領域内の各ブロックと、該ブロックに隣接するブロックに対しては上記離散コサイン変換処理後の得られる係数の所望の一部を所望値に変更処理する処理手段とを設ける。また、処理手段には、画像を前記指定された局所領域に従って分割する第1の機能、この第1の機能により分割された領域内の画像について上記画像ブロック単位で離散コサイン変換処理すると共に、上記指定された各小ブロックと、該画像ブロックに隣接するブロックに対しては上記離散コサイン変換処理後の得られる係数の所望の一部を所望値に変更処理する第2の機能とを設ける

【0020】

【作用】上記第1の構成の場合、画像の或る領域をなすブロックのデータが編集或は圧縮・伸張等の操作により元のデータからの歪みを生じる場合に、そのブロックを離散コサイン変換して得た係数の一部を変更する事によって、そのブロック領域と隣接する画像領域との間の歪みを除去或いは所要許容誤差の範囲内に抑えることができる。

【0021】また、処理手段に、係数変更の際に、直流成分に対応する係数及び交流成分のうちの最も低い周波数を持つ基底関係に対応する係数を所望値に変更処理する機能を持たせる構成とすると、画像の或る領域をなすブロックのデータが編集或は圧縮・伸張等の操作により元のデータからの歪みを生じる場合に、そのブロックを離散コサイン変換して得た係数の一部を特に直流成分に対応する係数及び最も低い周波数を持つ基底関係に対応する係数を変更する事によって、そのブロック領域と隣接する画像領域との間の歪みを除去或いは所要の許容誤差の範囲内に抑えることができる。

【0022】第2の構成の場合、画像の或る領域をなすブロックのデータが編集或は圧縮・伸張等の操作により元のデータからの歪みを生じた場合に、ブロック内の境界領域を含む一つ以上の小ブロックを指定し、それらの各小ブロック毎に請求項3に記された手段を用いて、該ブロック領域と隣接する画像の領域との歪みを除去する或はその歪みを或る許容誤差の範囲内に抑えることができる。

【0023】第3の構成の場合、画像の編集に携わるオペレータは画像データに対して、画像の局所領域とその領域における編集要求を指定する。すると画像データはその指定領域に従ってブロックに分割され、各ブロックは編集要求に従ってデータの変更がなされる。この変更により各ブロックの境界に歪みが生じる場合があり、そ

れを低減するために、そのブロックを離散コサイン変換して得た係数の一部を特に直流成分に対応する係数及び最も低い周波数を持つ基底関係に対応する係数を変更する処理を行う。この処理によって、そのブロック領域と隣接する画像領域との間の歪みを除去或いは所要の許容誤差の範囲内に抑えられるので、画像をオペレータの指定に従って局所領域毎に編集できると共に、編集後に領域間の歪みが生じないように出来る。

【0024】第4の構成の場合、画像を圧縮して蓄積する作業に携わるオペレータは画像データに対して画像の局所領域とその領域における圧縮要求を指定すると、画像データはその指定領域に従ってブロックに分割され、各ブロックが圧縮要求に従って圧縮され蓄積される。種々の圧縮要求の中で、特にブロック歪みの低減及び圧縮後のブロックのビット数の低減の要求に対しては、画像ブロック内を所望の小ブロックに分割すべく指定すると共に、上記画像ブロックの境界領域を含む一つ以上の小ブロックを指定する。すると、上記画像ブロック単位で離散コサイン変換処理され、上記指定された各小ブロックと、該画像ブロックに隣接するブロックに対しては上記離散コサイン変換処理後の得られる係数の所望の一部を所望値に変更処理する。これにより、ブロック歪みの低減及び圧縮後のブロックのビット数の低減の要求に対して、ブロック歪みのない画像再生ができるように補正して圧縮できる。

【0025】更には、画像データを編集し、編集結果を圧縮して蓄積する際に、作業に携わるオペレータは画像の編集にあたって、第2の機能を設けたことにより、第1の機能にて分割された領域内の画像について上記画像ブロック単位で離散コサイン変換処理すると共に、上記指定された各小ブロックと、該画像ブロックに隣接するブロックに対しては上記離散コサイン変換処理後の得られる係数の所望の一部を所望値に変更処理し、これを圧縮処理して蓄積する結果、オペレータの指定する領域毎の編集を行い、編集後の境界領域の歪みを除去し、その結果を効率的に圧縮することが出来る。

【0026】

【実施例】以下、本発明の実施例について、図面を参照して説明する。

(実施例1) 実施例1は、画像の或る領域をなすブロックのデータが、編集或いは圧縮・伸張等の操作により元のデータに対する歪みを生じた場合に、そのブロックを離散コサイン変換して得た係数の一部を変更することによって、そのブロック領域と隣接する画像領域との間の歪みを除去するか、或はその歪みを或る許容誤差の範囲内に抑えるようにして、そのブロック領域と隣接する画像領域との間の歪みを除去若しくはその歪みを或る許容誤差の範囲内に抑えることができるようにしたものである。

【0027】また、上述した係数変更の際に、直流成分

に対応する係数及び最も低い周波数を持つ基底関係に対応する係数を用いるようにする。図1は本発明の一実施例を示すブロック図である。図1において、1は操作部、2はCPU（プロセッサ）、3はバス、4は入力I/F（インタフェース）、5はプログラムメモリ、6はワークメモリ、7は画像メモリ、8は出力I/F（インタフェース）、9は外部記憶装置、10はGDC（グラフィック・ディスプレイ・コントローラ）、11はディスプレイである。

【0028】これらのうち、操作部1は、システムに操作入力を与えたり、指令を与えたりするためのマンマシンインタフェースであって、キーボードやタッチパネルあるいはポインティング・デバイスなどが相当する。そして、画像データの編集の際に、その画像の編集に携わるオペレータが、画像の領域を指定するにあたってはマウス、ペン、レーザー・ポインタ、ジョイスティックなどのリモート・コントロールによるポインティング・デバイスを使って画像の領域を指定する。

【0029】CPU 2は、プログラムメモリ5に記憶されたプログラムに従い、画像の領域分割等の処理を実行するものである。入力I/F 4は、処理対象となる画像データを本画像処理装置に入力するためのインタフェースである。画像データは自然画像でも、また、人口的に作成されたグラフィックスなど、何でも構わない。また、1画面はこれも限定されるものではないが、例えば512×512画素のRGB（但し、Rはレッド、Gはグリーン、Bはブルー）（またはCMYK（但し、Cはシアン、Mはマゼンタ、Yはイエロー、Kはブラック）、あるいはCIE LAB）のカラーのデータより構成される。

【0030】プログラムメモリ5は、上述したCPU 2の処理手順を表わすプログラムデータを格納するためのものであり、CPU 2はこのプログラムメモリ5の格納プログラムを実行して目的の処理を行う。

【0031】ワークメモリ 6はCPU 2が領域分割の処理に伴い算出するデータ（後述するツリーデータ、リストデータ等）を一時保持しておくためのものである。また、画像メモリ7は入力I/F 4から入力された画像データを蓄えるためのものである。

【0032】出力I/F 8は、領域分割された画像データ等を外部機器（プリンタ等）に出力するためのデジタルインタフェースであり、また、外部記憶装置9は、ハードディスクユニット、光磁気ディスクユニット等により構成され、画像データ、領域分割された画像データ、ツリーデータ、リストデータ、プログラムデータ等を保存する大容量記憶装置である。

【0033】GDC 10は画像メモリ7上の画像データを、画像表示用のデータに変換して出力するものであり、ディスプレイ11はGDC 10の出力する表示データに基づく画像を表示してオペレータに提示するため

のものである。

【0034】本実施例に係る画像処理の詳細を説明する。CPU 2は画像データを入力I/F 4を介して、あるいは外部記憶装置9から取り込む。CPU 2は取り込んだ画像データを画像メモリ7に記憶させる。画像メモリ7に記憶された画像はGDC 10により画像表示データに変換され、ディスプレイ11に与えられて画像として表示される。

【0035】つぎにオペレータは画像の編集を行うべく、このディスプレイ11の表示画像を見て操作部1より画像の編集操作を行うと、CPU 2はプログラムメモリ5中の編集処理のためのプログラムを実行し、画像メモリ7の画像データに対して、オペレータの編集指示に従った編集処理を行う。そして、その編集処理した画像データを画像メモリ7に記憶させる。画像メモリ7に記憶された編集処理済み画像はGDC 10により画像表示データに変換され、ディスプレイ11に与えられて画像として表示される。

【0036】編集が終わると、オペレータは画像メモリ7上の当該編集処理済み画像のデータを外部記憶装置9に保存すべく操作部1より保存操作を行う。するとCPU 2はプログラムメモリ5中の保存処理のためのプログラムを実行し、画像メモリ7の画像データを外部記憶装置9に保存する。このとき、オペレータが画像圧縮保存を指定して画像保存を指示していたとすると、CPU 2は保存すべき画像データを、圧縮処理して保存する。その際の圧縮率は、標準の圧縮率か、またはオペレータ指定の圧縮率である。また、圧縮処理に使用する方式は国際標準方式である。そのため、取り込んだ1画面分の画像を所定の画素サイズのブロック（画像ブロック）に分割し、この分割した各ブロックに対してそれぞれ離散コサイン変換（DCT）してブロック内画像成分を直流成分と各周波数成分に分け、さらにこれを量子化してから、この量子化されたデータを可変長符号化して符号化データとした上で外部記憶装置9に保存する。

【0037】こうして画像圧縮処理されて外部記憶装置9に保存された圧縮画像は再生時には、その逆の手順を踏んで伸長され、元の画像に戻される。すなわち、CPU 2はプログラムメモリ5中の圧縮伸長処理のためのプログラムを実行し、外部記憶装置9から読み出された符号化データを、可変長復号化処理し、そして、逆量子化処理して後に逆DCTを行い、画像データを元のデータに伸長させる。そして、この伸長させた画像データを画像メモリ7に記憶させる。画像メモリ7に記憶された編集処理済み画像はGDC 10により画像表示データに変換され、ディスプレイ11に与えられて画像として表示される。

【0038】ここで、画像編集あるいは画像圧縮伸長をした結果、ブロック歪みが目立つようになったとする。オペレータはディスプレイ11に表示される画像からこ

れがわかるので、操作部1を操作してブロック歪み補正指令を与える。すると、CPU 2はプログラムメモリ5中のブロック歪み低減プログラムを実施して画像メモリ7上の画像データにおけるブロック歪み補正対象のブロックの画像データについて当該歪みを補正し、補正したデータに置き換える。そして、この置き換えにより、ブロック歪みの無い、もしくは低減された画像が得られるようになる。

【0039】ブロック歪み補正指令は、例えば、補正指令レベルを指定することで行う。補正指令レベルは段階指定例えば、0, 1, 2, 3, 4..., Nと云ったもので良く、1段階目(レベル1)を指定すれば、後述する変更量に関するベクトル e の要素 $e[1]$ だけ歪みを補正するようになり、2段階目(レベル2)を指定すれば変更量に関するベクトル e の要素 $e[1]$ と $e[2]$ だけ歪みを補正するようになり、3段階目(レベル3)を指定すれば、変更量に関するベクトル e の要素 $e[1]$ と $e[2]$ および $e[3]$ だけ歪みを補正すると云った具合に処理する。もちろん、変更量に関するベクトル e のN個ある要素のうち、所望とする要素 $e[i]$ (i は0,1,2,3,4,...,N)を幾つか個別指定する方式でも良い。

【0040】ブロック歪み補正について説明する。1画面の画像におけるブロック歪みが顕著になった或る領域を成すブロック(画像ブロック)に着目してみる。ここで、実施例1においては画像の当該領域をなすブロックが、1次元の配列(つまり、1行×Nカラムの画素構成)で表わされる場合についての例を示す。

$$e = [e[1], e[2], \dots, e[N]]^T \quad \dots (1)$$

なる変更量ベクトルを想定する。今、変更指令は $e[1]$ と $e[N]$ の分だけの指定であるから、 $e[1]$ 、 $e[N]$ だけ変更を要求されていることになる。そして、 $e[1]$ は $y[1]$ に対応する要素であり、 $e[N]$ は $y[N]$ に対応する要素であるから、結局、 $e[1]$ だけ $y[1]$ を変更し、 $e[N]$ だけ $y[N]$ を変更すれば良い。しかし、これらは未知数である。一般的には、 e は任意のベクトルで良い

$$f = [f[1], f[2], 0, 0, \dots, 0]^T \quad \dots (2)$$

とし、また、原画像のあるブロックを構成するベクトル f の要素のうち、 $f[1]$ 、 $f[2]$ が変数であるとして、

$$g = [g[1], \dots, g[N]]^T \quad \dots (3)$$

とすると、逆DCT処理後のベクトル g の各要素は $f[1]$ 、 $f[2]$ の関数となる。また、式(3)において、

$$g[1] = e[1] + \text{sgn}[e[1]] \varepsilon[1] \quad \dots (4)$$

$$g[N] = e[N] + \text{sgn}[e[N]] \varepsilon[N] \quad \dots (5)$$

とする。なお、 ε は許容誤差を示すベクトルであり、また、ここで、 $\text{sgn}[a]$ は a が正の時に“1”、負の時に“-1”、0の時に“0”の符号関数である。

【0046】この式(4)および式(5)なる2つの式は $f[1]$ と $f[2]$ と云う2つの変数を持つ線形一次方程式であるから、これを解くことができ、要素 $f[1]$ 、 $f[2]$ を求めることができる。そして、これにより $f[1]$

【0041】この場合、画像の当該領域をなす各ブロックそれぞれの画素値が、1次元のベクトル $x = [x[1], x[2], \dots, x[N]]^T$ で表わされるものとする。 x は原画像におけるブロック分割された1つのブロック全体を示すベクトルであり、 $x[1]$ 、 $x[2]$... $x[N]$ はそれぞれ x の要素を示していて、 $x[1]$ は第1画素の、 $x[2]$ は第2画素の、...、そして、 $x[N]$ は第N画素の、それぞれ画素値を示すベクトルである。

【0042】ここで画素値は輝度と仮定する。但し、色差あるいは他の色空間の要素値であっても同様に処理できる。このブロックのベクトル x がオペレータによる編集作業の後、 $y = [y[1], y[2], \dots, y[N]]^T$ に変わったとする。そのとき、“ y ”の要素値のいくつかを変更すれば、原画像の画素値(特にブロックの境界領域の部分の画素値)に近づけたり、或は該ブロックに隣接するブロックとの間の歪みを少なくすることができる。

【0043】“ y ”の要素値の変更はオペレータが操作部1を操作してCPU 2に変更指令を与えることにより、CPU 2が、次のような理論に基づく変更処理を行うことでなされる。一般的な話をするため、ここでは変更指令は変更の対象を個別に指定することとする。そして、その対象として $e[1]$ と $e[N]$ を指定したとする。

【0044】ここで、“ e ”を変更量に関するベクトルとし、

から、以下の例を容易に応用することができる。

【0045】本例の場合、 $e[1]$ 、 $e[N]$ だけの変更であるから、変更量に関するベクトル e は未知の2つの特定要素である $e[1]$ と $e[N]$ を含む。そこで、長さNのベクトル f (原画像をブロック分割したときのブロックあたりの構成画素数が1行N列計N個構成である場合のベクトルで上記 x に相当する)を

上記のベクトル f に逆DCTを施した結果のベクトル g を

$g[1]$ 、 $g[N]$ は

、 $f[2]$ を要素とするベクトル f が求まることから、当該 f を逆DCTすることで、 g が求まり、この g を y から引いて、 $y - g$ を求めることで、該ブロックの歪み補正した新しい画素値が得られる。すなわち、上記ブロックを編集処理してそれを圧縮処理した後に、伸長処理して画像を復元した場合の歪み分を、補正するに必要な補正值分、予め補正したかたちの新しい画素値が得られ

る。

【0047】本発明ではこれを利用して、編集処理した圧縮前の画像について、その編集処理した部分での画像データについて、事前に歪み補正処理を施してしまうようにする。

【0048】CPU 2には図2に示すように、ソフトウェアにより、ブロック指定処理機能2a、ブロック編集処理機能2b、ブロック補正処理機能2cを持たせてあり、オペレータが操作部1の構成要素の一つであるポインティング・デバイス1aを操作して画像上の所望の領域を指定すると、原画像データに基づいてブロック指定処理機能2aはこの領域内の画像データのブロック情報を取り出す。

【0049】ここでブロック情報とは、画像圧縮する際に画像分割するが、その分割された場合の上記指定領域の該当ブロックの情報である。領域指定されると、編集処理が可能になる。オペレータは操作部1を操作して所望の編集要求を行う。すると、CPU 2はこの要求に基づいてブロック編集処理機能2bにより、編集を行う。この編集は上記のブロック情報と原画像データに基づいて原画像データにおける上記指定領域の該当ブロック対応の画像データについて、“ぼかし”や“シャープネス向上”、“明るさ”や“色の調整”等々指定された編集処理を施すかたちで行う。この編集処理により、編集された画像データと編集により変更されたブロックの情報が生成される。つまり、編集により、あるブロックでの原画像のデータによるベクトル x は、 y になる。

【0050】補正処理が施される前の段階では、CPU 2は編集処理した画像のデータを画像メモリ7に書き込むので、この画像メモリ7の画像データをGDC 10で画像表示用のデータに変換し、ディスプレイ11に表示させる結果、編集された状態の画像を表示してオペレータに提示することになる。

【0051】この画像を見てオペレータはつぎに操作部1の操作により、補正要求を出す。CPU 2はこの補正要求により、編集処理済みブロックの画像データに対して、ブロック補正処理機能2cによるブロック補正処理を施す。

【0052】ここでのブロック補正処理は上述した原理に基づく処理であり、編集処理済みブロックの画像データ(上述の y に相当)に対して、上述の g を求め、この g を y から引いて、 $y-g$ を求める処理をすることで、該ブロックの歪み補正をした新しい画素値を得る。この補正により、上記指定領域の該当ブロックを編集処理してそれを圧縮処理した後に、伸長処理して画像を復元した場合の歪み分を、補正するに必要な補正值分、予め補正したかたちの新しい画素値が得られる。

【0053】CPU 2は編集処理後の画像について、編集対象となったブロックの画素値をこの補正された画素値に置き換え、最終的な画像データとする。そして、

保存したり伝送する場合に、この最終的な画像データを圧縮処理する。そして、この圧縮処理された画像を外部記憶装置9に記憶し、また、伝送する場合には出力I/F 8に送り出す。

【0054】この結果、圧縮画像を伸長した場合に、歪みの補正された、あるいは歪みの少ない画像を復元することができる。本システムでは、GDC 10は画像メモリ7上の画像データを、画像表示用のデータに変換して出力し、ディスプレイ11はGDC 10の出力する表示データに基づく画像を表示してオペレータに提示するので、オペレータは編集処理済みの画像を観察することができ、その編集処理済みの画像におけるブロック歪みの目立つところあるいは圧縮伸長後にブロック歪みが懸念されるところ等を操作部1にて指示するだけで、歪みを補正処理するから、この画像を圧縮後、伸長した場合に、ブロック歪みのない画像が観察できるようになる。

【0055】なお、この例ではベクトル f の要素のうち、変数となる要素を $f[i]$ 、 $f[j]$ と表わすようにしたが、一般的には $f[i]$ 、 $f[j]$ でよい。但し、 $1 \leq i \leq N$ 、 $1 \leq j \leq N$ 、 $i \neq j$ である。

【0056】このように実施例1は、画像の或る領域をなすブロックのデータが、編集或いは圧縮・伸張等の操作により元のデータに対して歪みを生じる場合に、そのブロックを離散コサイン変換(DCT)して得た係数の一部を変更することで、画像データを補正する。この変更によって、そのブロック領域と隣接する画像領域との間の歪みを除去、あるいは或る許容誤差の範囲内に抑えることができるようになる。

【0057】その結果、そのブロック領域と隣接する画像領域との間の歪みを除去若しくは或る許容誤差の範囲内に抑えることができるようになり、画像の品位を保つことができるようになるものである。

【0058】さらにこの実施例1では、上述した係数変更の際に、直流成分に対応する係数及び最も低い周波数を持つ基底関係に対応する係数を用いるようにすると、歪みを除去若しくは低減することができるようになる。つまり、画像の或る領域をなすブロックのデータが編集或は圧縮・伸張等の操作により元のデータからの歪みを生じた場合に、そのブロックを離散コサイン変換して得た係数の一部を特に直流成分に対応する係数及び最も低い周波数を持つ基底関係に対応する係数を変更する。この変更によって、そのブロック領域と隣接する画像領域との間の歪みを除去、あるいは或る許容誤差の範囲内に抑えることができるようになる。

【0059】実施例1において上述した“係数変更の際に、直流成分に対応する係数及び最も低い周波数を持つ基底関係に対応する係数を使う”場合の説明をする。ブロック歪みに最も大きな影響を与えているのが、低周波数成分である。

【0060】従って、低周波数成分について変更を行うようにする。このケースの最適な例は直流成分の基底関数であるDCT係数の $f[1]$ と、最も低い周波数成分の基底関数であるDCT係数の $f[2]$ に対して変更を加える場合である。

【0061】補正をするのに直流成分 $f[1]$ と最も低い基底関数のDCT係数 $f[2]$ を使う好適な例は、空間的にゆるやかな補正をする必要な場合であり、これはグラデーションのある画像領域のブロックに対して適用するとより効果が高い。

【0062】ここで、長さ N のベクトル f （ブロックの画素構成が $1 \times N$ 画素での原画像のあるブロックでの画素値に相当）と、変更量に関するベクトル e に関する操作を、整理して具体的に説明する。ここでの処理は、ブロック補正処理部2cが担う処理である。

【0063】まず、DCTの操作はDCT対象に対して

$$f[i] = \sum_{j=1}^N m[i, j] (e[j] + \varepsilon'[j]) \quad \dots (8)$$

【0066】そして、

$$f' = [f[1], f[2]]^T \quad \dots (9)$$

$$e' = [e[1], e[2]]^T \quad \dots (10)$$

【0067】

$$M' = \begin{bmatrix} m[1,1] & m[1,N] \\ m[N,1] & m[N,N] \end{bmatrix} \quad \dots (11)$$

【0068】

$$\varepsilon'' = [\varepsilon'[1], \varepsilon'[N]]^T \quad \dots (12)$$

とおくと、 f' は

$$f' = M' (e' + \varepsilon'') \quad \dots (13)$$

となり、式(9)および式(13)からわかるように、要素 $f[1]$ 、 $f[2]$ は、2次元のベクトルの加算と 2×2 の行列のかけ算で求めることができる。こうして求められた f に逆DCTを施してその結果のベクトル g を求める。このベクトル g は、先の式(4)である

$$g[1] = e[1] + \text{sgn}[e[1]] \varepsilon[1]$$

と、式(5)である

$$g[N] = e[N] + \text{sgn}[e[N]] \varepsilon[N]$$

とを満たしている。 $g[i]$ （但し、 $i \neq 1$ 、 $i \neq N$ ）は“0”ではないが、 g はDCTを施した時、直流成分及び最も低い周波数成分しか含まないので、空間的にゆるやかな変化を示している。

【0069】このベクトル g をベクトル y から差し引いて得られた $y - g$ の各要素を新しく該ブロックの画素値とする。これによって、ブロック $[y[1], y[1], \dots, y[N]]^T$ で、そのうちの $y[1]$ に $e[1]$ の歪みがあり、 $y[N]$ には $e[N]$ の歪みがあった時の当該歪みを、DCT係数の変更によって補正することができた。

$$f' = (M'^T)^T (e' + \varepsilon'') \quad \dots (14)$$

（ここで、 $(M'^T)^T$ は M'^T の一般逆行列であ

る。）とおくことができることから、この式(14)を

$N \times N$ の行列 $M = [m[i, j]]$ 、 $i = 1, \dots, N$ 、 $j = 1, \dots, N$ をかけることであり、逆DCT(IDCT)の操作はDCTされたものに対して上記行列 M の転置行列 M^T をかけることであることに留意する。

【0064】行列 M はユニタリ性があるので、行列 M の逆行列 M^{-1} は上記転置行列 M^T と等しい。従って、上記行列 M の転置行列 M^T を式で表わすと

$$M^T f = e + \varepsilon' \quad \dots (6)$$

となる。但し、 $\varepsilon' = [\text{sgn}[e[1]] \varepsilon[1], \dots]^T$

である。行列 M と逆行列 M^{-1} を乗じたものは“1”（つまり、 $M \times M^{-1} = 1$ ）であるので、式(6)の両辺に M を乗じてベクトル f を表わすと、

$$f = M (e + \varepsilon') \quad \dots (7)$$

となる。これをベクトル f の要素を用いて表わすと、

【0065】

【数1】

20

【数2】

30

【0070】特に、この例では、直流成分の要素に関連する変数 $f[1]$ と、最も低い周波数成分の要素に関連する変数 $f[2]$ を用いているため、空間的にゆるやかな補正を施すことが出来た。

【0071】なお、この例では $e[1]$ 、 $e[N]$ という2つの補正要素に対して $f[1]$ 、 $f[2]$ という2つの変数で対応したが、一般にはブロックの要素数である N 以下の任意の正数の補正要求に対して、 N 以下の任意の正数の変数を対応させても一般逆行列を用いて上述の式(6)である

$$M^T f = e + \varepsilon'$$

を解くことができる。すなわち、先の実施例にならって、 f をDCTしたものを、逆DCTする場合、行列 M の転置行列 M^T を f に掛け算することであるから、原画像における分割したブロック内の成分を表わすベクトルである f における $f[1]$ 、 $f[2]$ の要素のみで表わされる成分のベクトル f' を求めるには、 $M^T f = e + \varepsilon'$ の時、 $M^T f' = e' + \varepsilon''$ として

解けば良い。一般逆行列の場合、実際の計算では誤差が生じるので、その分の誤差は許容誤差を定める時に推測

50

して考慮するようにする。

【0072】また、ブロックが2次元の場合には、まず横に1次元のブロックに分けて補正を行い、次に縦のブロックに分けて補正を行えば良く、また、横と縦の順が逆でも良い他、N次元の時に同様に実施できる。

【0073】以上の処理がCPU 2におけるブロック補正処理部2cで行われるが、このような処理機能を機能ブロックで示すと図3の如きとなる。すなわち、原画像の編集処理を行った結果、画像に歪みが見られる場合に、オペレータは操作部1を操作して補正要求を行うことになる。オペレータが補正要求を行うと、CPU 2は編集された画像データyと原画像データとを取り込み、歪みデータ生成処理を行う歪みデータ生成部21の機能によってここで両画像データから歪みデータe'を求める。また、補正要求と共に与えられる許容誤差の指示情報から許容誤差生成部22の機能によりε''を求める。そして、求めたe'とε''を加算処理部23により加算処理する。一方、基底関数処理を行う基底関数処理部24の機能によりM'を得、これを逆行列計算部25の機能により逆行列計算してM'^{-1}を得、これを加算処理部23による加算結果と共に乗算する乗算部26の機能により乗算すると、編集された画像データは歪み補正された画像データとなって得られる。

$$\begin{aligned} & e[1,1], e[1,2], \dots, e[1,N], \\ & e[2,1], e[3,1], \dots, e[N,1], \\ & e[N,2], e[N,3], \dots, e[N,N], \\ & e[2,N], e[3,N], \dots, e[N-1,N] \end{aligned} \quad \dots (15)$$

が補正されるように要求されているとする。この場合、変更量に関するベクトルeは4N-4個の任意でない要

$$\begin{aligned} & f[1,1], f[1,2], \dots, f[1,N], \\ & f[2,1], f[2,2], \dots, f[2,N], \\ & f[3,1], f[4,1], \dots, f[N,1], \\ & f[3,2], f[4,2], \dots, f[N,2] \end{aligned} \quad \dots (16)$$

と表わされ、f[1,1] から f[N,2] までの4n-4個の変数を持つが、それ以外の要素値は“0”であるとする。ベクトルfに逆DCT (IDCT)を施したものをqとする。すなわち、ベクトルfに逆DCTを施した結果のベクトルgの各要素は、ベクトルfの中の2N-4個の変数の関数である。

$$\varepsilon' = \begin{bmatrix} \text{sgn}[e[1,1]] \varepsilon[1,1] & \dots & \text{sgn}[e[1,N]] \varepsilon[1,N] \\ \vdots & & \vdots \\ \text{sgn}[e[N,1]] \varepsilon[N,1] & \dots & \text{sgn}[e[N,N]] \varepsilon[N,N] \end{bmatrix} \quad \dots (17)$$

行列の要素を用いて示すと

【0081】

【0074】以上、実施例1はブロックが2次元の配列で表わせる場合について、1次元の配列に分けて処理するようにしたものであるが、2次元行列で処理することもできるので次にその例を実施例2として説明する。

【0075】(実施例2) 実施例1の発明におけるブロックが2次元の配列で表せる場合について、実施例1では1次元の配列に分けて処理した例を示したが、ここでは2次元行列で処理する例を示す。

【0076】画像中の或る領域における、あるブロックでの構成画素の画素値がN×Nの行列x = [x[i,j], i = 1, ..., N, j = 1, ..., N] で表わされるとする。この場合にも、xが実施例1の解法を2次元へ拡張することができることを示す。

【0077】オペレータによる編集作業の後、上記xがN×N行列yに変わったとする。そのとき、yの要素値のいくつかを変更して原画像の画素値x (特にブロックの境界領域の部分) に近づけたいか或いは該ブロックに隣接するブロックとの間の歪みを少なくしたい等の目的で (つまり、ブロック歪み低減のために) オペレータはyを変更する補正指令を操作部1より与えたとする。ここで、eを変更量に関するM×Nの行列とする。

【0078】この行列では、オペレータからの補正指令によってブロックの境界部分である

素を含むことになる。そこで、4n-4個の変数を持つN×Nの行列を想定する。このN×Nの行列は、

【0079】実施例1と同様に、
M' f M = e + ε'

但し、

【0080】

【数3】

【数4】

$$\sum_k \left(\sum_j m[i,j] f[j,k] \right) m[l,k] = e[i,1] + \epsilon' [i,1] \quad \dots(18)$$

即ち、

$$\sum_k \sum_j m[i,j] m[l,k] f[j,k] = e[i,1] + \epsilon' [i,1] \quad \dots(19)$$

これは、(j, k) のインデックスの代わりに s を、そして、(i, 1) のインデックスの代わりに t を用いて表わすと

$$\sum_s m' [t,s] f[s] = e' [s] + \epsilon' [s] \quad \dots(20)$$

の形になる。すなわち、

$$M' f = e' + \epsilon' \quad \dots(21)$$

の形になる。これは、実施例1と同じ形である。但し、ここでは $(M')^{-1} \neq M'^{-1}$ であるから、一般には一般逆行列 $(M')^{-1}$ を使う。この列の場合には、 $4N-4$ 個の変数と同数の方程式があるので、 M' のランクが f

$$g = M'^{-1} f$$

であり、この g を y から引いて、 $y - g$ の各要素を新しく該ブロックの画素値とする。なお、一般逆行列の場合、実際の計算では誤差が生じるので、その分の誤差も考慮して許容誤差を定めることが望ましい。

【0083】以上、実施例1および実施例2は、画像の或る領域をなす画像ブロックのデータが、画像における編集した領域、或いは圧縮・伸張等の操作により、原画像に対して歪みを持つような場合に、その領域の画像ブロックについて、そのブロックを離散コサイン変換(DCT)して得た係数の一部を変更することで、画像データを補正する。特に、DCTして得た係数のうち、変更対象の周波数成分を特定してその周波数成分要素の係数について所要の補正をすることにより歪みを低減するようにしたものである。そして、これにより、編集や画像圧縮しても歪みのない画像が得られるようになるものである。

【0084】本発明は、このようなブロック単位での補正方法以外にも原画像に対して歪みのある領域の画像ブロックにおけるそのブロック内部の小領域を直接指定してこの小領域およびこれと隣接する領域について歪み補正することにより、処理時間を短縮して、しかも、実施例1および実施例2と同様に、歪みの無いあるいは歪みの低減された画像を得ることができるようになることができる。その例をつぎに説明する。

【0085】(実施例3) 実施例3は、画像の或る領域をなすブロックのデータが編集或は圧縮・伸張等の操作により原画像に対して歪みを生じる場合に、歪みを生じる画像ブロック内の境界領域を含む一つ以上の小ブロックを指定し、それらの各小ブロック毎に実施例1または実施例2の手法を用いて補正処理することにより、該ブロック領域と隣接する画像の領域との歪みを低減することができるようにするものである。

ルであると仮定すれば、 M' の逆行列 $(M')^{-1}$ が存在し、従って、 $(M')^{-1} = (M')^{-1}$ である。このように、 f が求められたので、それを逆DCTにより変換してその結果のベクトル g を求めると、

$$\dots(22)$$

【0086】これにより、画像の或る領域をなすブロックのデータが、編集或は圧縮・伸張等の操作によって、原画像に対して例えばブロック歪みを生じた場合に、当該ブロック歪みを生じたブロック内の境界領域を含む一つ以上の小ブロックを指定するだけで、歪みの生じたブロック領域とこれに隣接する画像領域との間の歪みを除去したり、或いはその歪みを或る許容誤差の範囲内に抑えたりして軽減することが短時間でできるようにする。

【0087】実施例3の詳細を説明する。図4(a)に示すように、1画面の画像中の或る領域をなす画像ブロックをBとし、この画像ブロックBの中の小領域である小ブロックを B_i (但し、 $i=0, 1, 2, \dots, 8$) とする。小ブロック $B_1 \sim B_8$ は画像ブロックBの境界領域にあり、小ブロック B_0 は中央にある。従って、小ブロック B_0 以外は画像ブロックBの境界を含んでいる。ここで図4(b)に斜線で示した領域である画像ブロックBの境界部分b u aの領域に生じたデータの歪みを補正する要求があるとする。

【0088】この要求に対して、画像ブロック内の境界領域を含む一つ以上の小ブロックを指定し、それらの各小ブロック毎に実施例1或いは実施例2で説明した手法を適用して、該ブロックBの境界部分b u aおよびこの領域と隣接する画像領域の歪みを補正することができる。ここで、画像ブロックBが比較的大きい場合には、図4(a)のように画像ブロックB内に小ブロック $B_0, B_1 \sim B_8$ をつくり、これらのうち、各小ブロック $B_1 \sim B_8$ に実施例1或いは実施例2で説明した手法を適用して補正することができる。

【0089】つまり、小ブロックは $B_0, B_1 \sim B_8$ の計9個あるが、図4(a)の小ブロック B_0 には歪み補正の要求がないので、小ブロック $B_1 \sim B_8$ だけにのみ補正する。そして、この場合、ブロックB全体みについて処

理せずに済み、小ブロック B0 の処理が不要となる分、効率的な補正処理をすることができるようになる。

【0090】(実施例4) この実施例4は、画像データの編集の際に、その画像の編集に携わるオペレータが1画面の画像に対してその画像の編集したい局所領域の指定と、その領域における編集要求をすると、画像データをその指定領域対応にブロック分割し、この分割した各ブロックについて、編集処理をすると共にこの編集処理をした領域について実施例3の処理を施すことにより、上記分割された各ブロックの境界に生ずる歪みを低減する為の補正をも自動的になされるようにしたものである。

【0091】これによって、画像はオペレータの指定に従って局所領域毎に編集され、しかも編集後に領域間の歪みが生じない或は許容誤差の範囲内に抑えられるようになるものである。

【0092】実施例4に詳細を説明する。画像データの編集の際に、その画像の編集に携わるオペレータが、操作部1を構成するポインティング・デバイス例えば、マウス、ペン、レーザー・ポインタなどのリモート・コントロールによるポインティング・デバイスを使って画像の領域をA～Eまで指定する。この指定情報はCPU2によって認識され、CPU2はこのポインティング・デバイスからの入力情報に対応して、画像メモリ7上の画像を図5のように、BA～BEにブロック分割する。ブロックの境界は明確にわかるように、画像メモリ7上の画像に対して分割ラインのデータを書き込む。

【0093】画像メモリ7に記憶された画像はGDC10により画像表示データに変換され、ディスプレイ11に与えられて画像として表示されるので、オペレータは分割ラインがわかる。

【0094】つぎにオペレータはこの画像の編集を行うべく、このディスプレイ11の表示画像を見て操作部1より、画像の編集操作を行う。この編集操作は、各ブロックBA～BEの境界をオペレータが操作部1を構成するポインティング・デバイスを操作することにより、ディスプレイ11の画面上で指示し、所望の編集要求をすることでCPU2がこれを認識し、この要求に従って画像メモリ7に記憶された画像のデータに対して実施例1または実施例2の手法を各指定ブロックに対して実施例3のように施すことで行われる。

【0095】画像メモリ7上の画像に対して歪み補正処理を行うと、この補正処理された画像はGDC10により画像表示データに変換され、ディスプレイ11に与えられて画像として表示されるので、オペレータは補正後の画像(歪みのない画像)を観察できるようになる。また、オペレータは操作部1を操作してこの画像メモリ7上の画像データを外部記憶装置9に保存したり、あるいは出力I/F8より外部の記録装置等に送り出すことで、ブロック歪みのない画像を記録できる。

【0096】この実施例では、画面上の画像について、オペレータが所望の領域分けを指定すると、その指定に基づく領域に画像をブロックに分割でき、分割した各ブロックに対して、そのブロックに対する編集要求を行うと、その要求に従った編集を自動的に行うことができる。編集要求には、各種フィルタリングによる“ぼかし”や“シャープネス向上”、“明るさ”や“色の調整”等々、様々なものがある。その編集要求に従って、CPU2は各ブロックの編集を行い、データが変更される結果、編集後の画像データには、境界領域のブロック内に歪みが生じるが、本実施例では、補正したい位置を画面上で位置指定することで、この指定位置のブロックについてCPU2が実施例3の手法を実施することで、編集により境界領域のブロック内に生じているこの歪みを自動的に補正することができるようになり、操作性が著しく向上する。

【0097】以上実施例1乃至実施例4は、1画面の画像に対して、所要の領域を指定して、その領域内の画像ブロックについて歪み補正を施すようにしたものであり、ブロック歪みが目立ちそうな領域について歪み補正を施すことに主眼があった。しかし、このようなケースの他に、画面の所望の領域について例えば、他の領域と圧縮率を変えて画像を圧縮する等の要求もある。これは例えば、背景像と注目被写体像と云った具合に、背景像は高圧縮率で、注目被写体像は画質劣化のないように比較的低下圧縮率で圧縮したいと云った場合や、また、注目被写体像部分でも特にその一部の領域は比較的低下圧縮率で圧縮し、他は高圧縮率で圧縮したいと云った要求のある場合である。

【0098】このような要求に応える実施例をつぎに説明する。

(実施例5) 実施例5は、画像データを圧縮して蓄積する際に、その画像の圧縮処理に携わるオペレータが、画像データに対して画像の局所領域の指定と、その領域における圧縮率の指定等を含めた圧縮要求を行うと、画像データがその指定領域に対応してブロックに分割され、各ブロックが圧縮要求の内容に従って圧縮され、蓄積され、圧縮による歪みを低減する処理が実施例3の手法によってなされ、それによってオペレータの要求にあった効率的な画像データの圧縮を可能にするものである。

【0099】実施例5の詳細を説明する。画像データを圧縮して蓄積する際に、その画像の圧縮に携わるオペレータが画像データに対して、実施例4にあるように操作部1を操作して画面中の局所領域を指定すると共に、この局所領域での画像データの圧縮要求の指示内容を指定する。ここでは実施例と異なり、局領域は編集対象ではなく、圧縮要求の指示内容に基づく処理をさせるための対象領域である。

【0100】圧縮要求の指示内容としては、例えば圧縮率の指定、低周波数領域強調、エッジ強調等々、種々の

ものがある。このような圧縮要求を操作部 1 の操作により行くと、CPU 2 はつぎのような処理を実施する。すなわち、画像の局所領域が指定されることで、1 画面分の画像ではその局所領域が他と分割されてブロック化される。このブロック化により、図 4 のようなブロックが生成されたとする。そのとき、各ブロックに対してオペレータから圧縮要求が指定される。圧縮要求としては、例えば圧縮率の高低の指定、低周波数領域強調、エッジ強調等々、種々のものがある。

【0101】静止画像や動画の圧縮手法のうち、J P E G や M P E G によって代表される圧縮手法では画像を 8 × 8 画素単位と云った小画像ブロックに分割して処理する。その小画像ブロック毎に圧縮を行うとその処理の際に、量子化による量子化誤差がブロック歪みやモスキート・ノイズの形で発生することがある。このような歪みを低減するために、実施例 3 の手法を利用する。

【0102】すなわち、各小画素ブロックの歪みを行列 e として、この e の中のいくつかの要素を許容誤差内に抑えるように、該小画素ブロックに D C T を施した時の係数のいくつかを変更する。そして、その変更に使われた D C T 係数を量子化するための数は、量子化誤差が許容誤差内に収まるように小さく抑えておく。

【0103】実施例 3 の手法を適用するにあたっては、 x 、 y 、 e 、 f はつぎのように対応させる。例えば、 x は小画素ブロックのデータであり、小画素ブロックのデータが $N \times N$ の行列である。そして、 $N \times N$ の行列 x を量子化したデータが行列 y であり、行列 y の D C T 係数が f 、その時の誤差が e である。そして、この場合、 e [1,1]、 e [1,N]、 e [N,1]、 e [N,N] を “0” にするように、実施例 1 の手法を用いて f [1,1]、 f [1,2]、 f [2,1]、 f [2,2] を変更する。つまり、直流成分の要素と、交流成分のうち最も周波数の高い成分の要素についての e が “0” となるように、データの変更処理を行う。このときの f [1,1]、 f [1,2]、 f [2,1]、 f [2,2] を量子化するための数は、許容誤差の 2 倍以下とすれば良く、このような数値を以て量子化すれば誤差を許容誤差内に小さくできる。

【0104】このように実施例 5 によれば、画面の所望の領域を指定してその領域について他と異なる圧縮率を指定したり、低周波数領域強調やエッジ強調等をする場合にも、歪みのない高品質の画像が再現できるようにな

る。なお、本発明は種々の実施例について説明したが、上述した実施例に限定されるものではなく、適宜変形して実施可能である。

【0105】

【発明の効果】以上述べたように、本発明によれば、編集や画像圧縮伸長処理において、問題になる歪みを解消できて高品位の画像を再現できる他、1 画面の画像における所望領域をブロックにより指定することができ、それらのブロック毎に指定された編集を施した後に生じるブロック間の歪みを許容誤差内に抑えることができるようになる。また、画像を圧縮する際に、圧縮率の高低やエッジ強調などの要求に従って画像をブロックに分け、それらのブロック毎に、J P E G や M P E G に代表されるブロック・ベースの圧縮手法において生じるブロック歪みなどの歪みを補正することができるようになる。

【図面の簡単な説明】

【図 1】本発明の実施例を説明するための図であって、本発明の一実施例の全体構成を示すブロック図。

【図 2】本発明の実施例を説明するための図であって、本発明の一実施例における要部構成の機能ブロック図。

【図 3】本発明の実施例を説明するための図であって、本発明の一実施例における要部構成の機能ブロック図。

【図 4】本発明の実施例を説明するための図であって、本発明の一実施例における処理例を説明するための図。

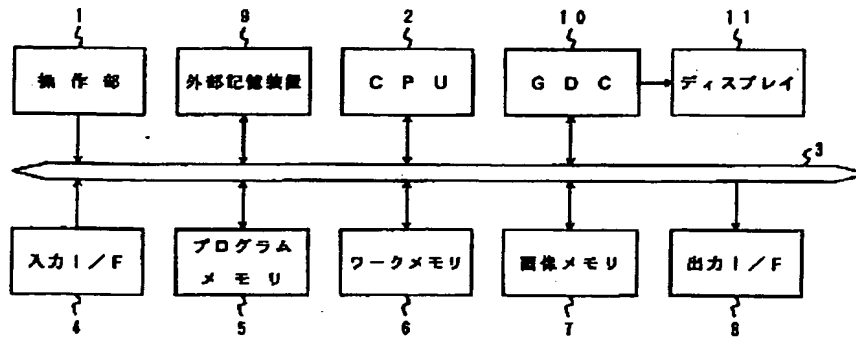
【図 5】本発明の実施例を説明するための図であって、本発明の一実施例における処理例を説明するための図。

【図 6】ブロック歪みを説明するための図。

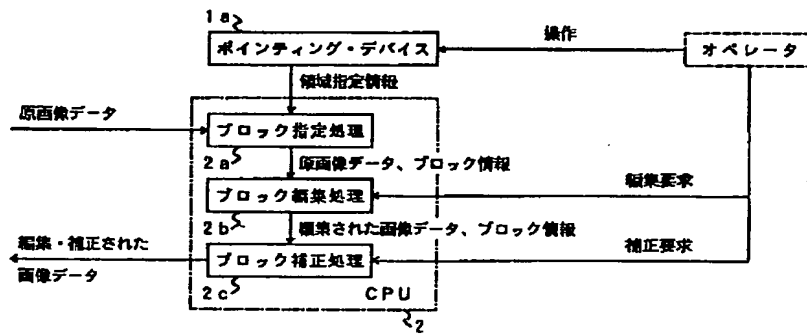
【符号の説明】

- | | |
|----|------------------------------|
| 1 | 操作部 |
| 2 | C P U (プロセッサ) |
| 3 | バス |
| 4 | 入力 I / F (インタフェース) |
| 5 | プログラムメモリ |
| 6 | ワークメモリ |
| 7 | 画像メモリ |
| 8 | 出力 I / F (インタフェース) |
| 9 | 外部記憶装置 |
| 10 | G D C (グラフィック・ディスプレイ・コントローラ) |
| 11 | ディスプレイ。 |

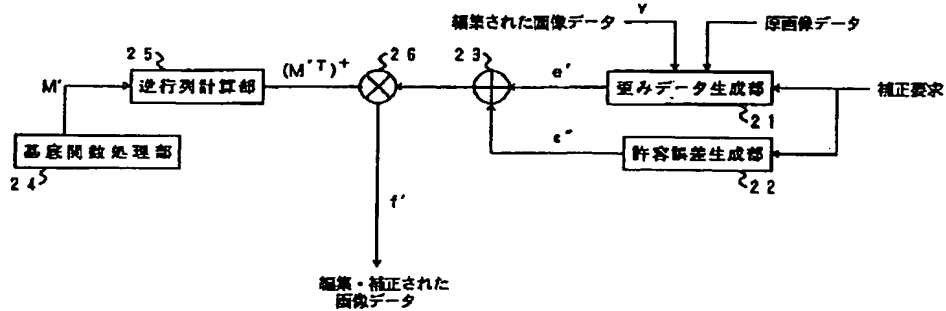
【図 1】



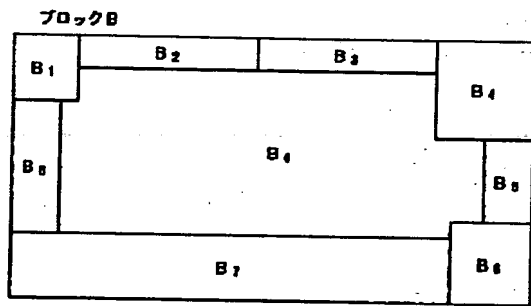
【図 2】



【図 3】

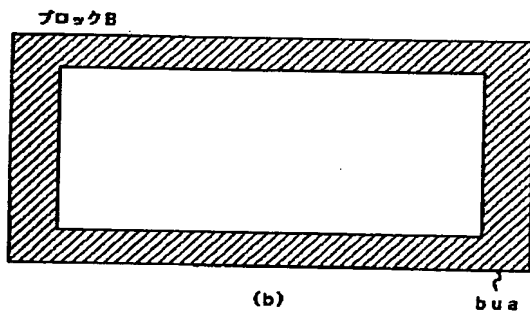
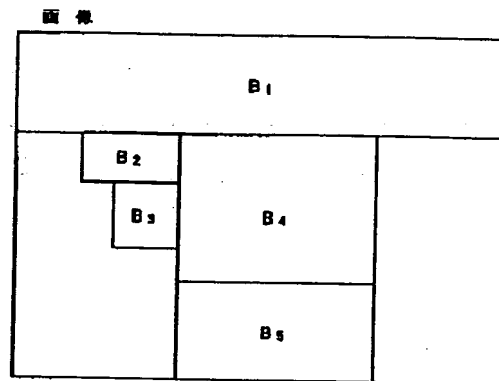


【図 4】



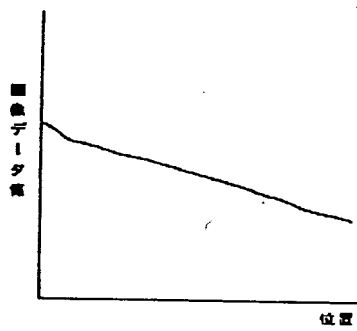
(a)

【図 5】

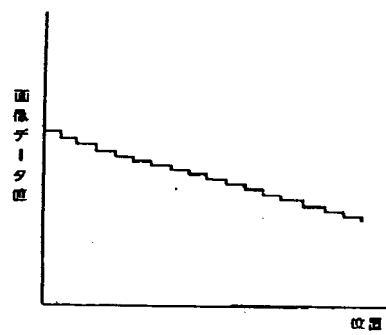


(b)

【図 6】



(a)



(b)